Metrics and methodologies for evaluating intelligent technologies

Emile Morse, Michelle Potts Steves, Jean Scholtz
National Institute of Standards and Technology
100 Bureau Drive
Gaithersburg, Maryland 20899, USA
{emile.morse, michelle.steves, jean.scholtz}@nist.gov

Keywords: software evaluation methodologies, user-centered evaluation, intelligent software

Abstract

In this paper we discuss the evaluation methodologies and metrics we have developed for ARDA's Novel Intelligence for Massive Data (NIMD) program. The thrust of this program is to produce intelligent software for the intelligence community. As such we are interested in producing metrics for the overall impact of the software on the analytic process, the analytic products, and the workload on the analyst. We are also interested in providing feedback to the researchers on the various types of intelligent software, user modeling and hypotheses generation, in particular. To accomplish this, we have developed metrics for the individual components and the overall program. In this paper we describe the process by which the metrics have been developed, the metrics, and the use of these metrics as software is introduced into the analysts' environment.

1. Introduction

Evaluation is a key component of the NIMD program (http://www.ic-arda.org/Novel_Intelligence/). Evaluation is needed to measure the progress of the program and to provide feedback to the researchers Additionally, metrics can be used to facilitate transition by helping potential customers assess quantitative measures of impact.

In this paper we discuss the process we used to develop the metrics, the metrics themselves, and the next steps in using the metrics. The NIMD program focuses on 5 research areas:

- Modeling Analysts and Analytical Processes
- Prior and Tacit Knowledge
- Hypothesis Generation and Tracking
- Massive Data
- Human Information Interaction.

The various research projects in this program had different approaches and tackled different combinations of the research areas. We felt it was important to have metrics that could measure and compare progress in each area. From the beginning we worked with the researchers and intelligence analysts to develop suitable measures that would be indicative of how the research software was working.

To this end we developed a metrics framework. This allows the conceptualization of a hierarchy of metrics and measures. That is, for each area of research, we can use the same metrics, but these may be implemented differently depending on the actual software. Therefore the measures may differ but the metrics remain the same. The next section details our Metrics Framework.

2. Metrics Framework

As we attempt to reuse metrics and measures, a mechanism for organizing the many and varied metrics and measures and their associated context is of value. We plan to use a *metrics model*, i.e., framework, developed at NIST for this purpose [Scholtz and Steves 2004, Steves and Scholtz 2005]. The metrics model provides a top-down approach for specifying system goal-directed software evaluations. When using the metrics model for evaluation design, system goals are mapped down through metrics and measures to direct collection efforts. Further the mappings assist in tying collected data back to their associated metrics and goal statements so that evaluation questions are answered during the evaluation analysis, as shown in Figure 1.

The upper levels of the framework are conceptual in nature, while the lower levels are reserved for application-specific feedback. This structure allows for the conceptual elements to be re-used in like-structured evaluations. Use of the metrics model in future evaluations has several envisioned benefits, namely: more re-use of metrics and measures for similar evaluations and the possibility of comparison of like-structured evaluations.

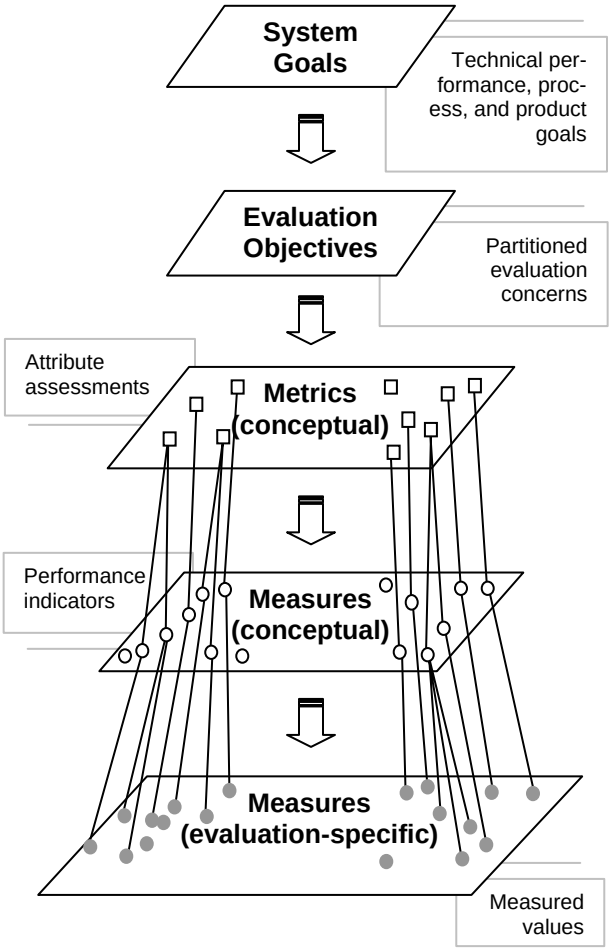

Figure 1: Metrics model

3. Process for Developing the Metrics

Early discussions and brainstorming sessions with the researchers gave us some ideas about the lower level measures that would be valuable to them to help in refining their approach.

As the research projects produced early versions of software components, we conducted pilot evaluations. For these evaluations, we were fortunate to have Naval Reservists as subjects. These reservists work as analysts for the Navy when on duty. Therefore, they were good "surrogates" for the analysts in the intelligence community, who we have limited access to because of the current demands of their jobs.

In order to conduct these evaluations, we also needed a way to easily collect data. The Glass Box, developed by Pacific Northwest Laboratories [Cowley, Nowell, and Scholtz, 2005] was used for this purpose. The Glass Box collects keystroke data, information about queries that are made, the web pages that are accessed, versions of documents that the analysts create, and annotation data. Analysts can insert comments about the software they are using, the analytic tasks they are performing, collaborations with others, and notes about which documents are relevant. The Glass Box software also collects video data of the screen and audio data from the subjects.

We used the Glass Box to collect data as we did pilot evaluations for the various research projects. These pilot projects were used to give the researchers feedback but also to develop the overall metrics. These metrics are discussed in the next section.

4. User Testing and Metrics

The second column of Table 1 shows the number of projects that have been evaluated compared with the total number of projects in each area.

Table 1: Number of projects evaluated by research area

NIMD Research Areas	#evaluated/total
Modeling Analysts and Analytical Processes (MAAP)	2/5
Prior and Tacit Knowledge (PTK)	1/4
Hypothesis Generation and Tracking (HGT)	3/7
Massive Data (MD)	1/5
Human Information Interaction (HII)	2/4

4.1 Metrics and measures used in evaluations

Table 2 summarizes the metrics and measures that were employed during evaluations in 2003 and 2004 for each NIMD research area. Note, the categories within each NIMD research area identified in the following table (e.g., Efficiency, Confidence) emerged after the evaluations had been designed, each addressing its specific evaluations requirements.

In addition to the above data, user questionnaire data were collected. From these we obtained demographic data and the users' perceptions of various aspects of the software.

Where indicated in Table 2, cognitive workload ratings were assessed by administering the NASA TLX (Hart and Staveland 1988). This survey asks subjects to rate their perceived levels of workload with respect to 6 scales — temporal, physical, mental, frustration, performance, and effort. Subsequently they indicate which factor was more important for each of 15 binary comparisons. The latter are used as weighting factors.

Finally, it is pertinent to note that many of these metrics were gathered by analyzing system logs and observation notes. At times observers captured timing information. At other times, Glass Box log data was analyzed for the timing information. Developers logged information from their systems as well that allowed us to capture additional data.

Table 2: Summary of Metrics by NIMD Research Area

Modeling Analysts and Analytic Process

Efficiency

- Time spent in the search tool
- Time on scenario
- Time spent in the tool compared with time spent in other applications
- Average time spent reading document when first viewed

Effort

- Number of queries made
- Number of links expanded
- Number of documents read
- Number of times each document was read
- Maximum depth of the link expansion

Accuracy

- Correctness of the answer

Confidence

- Relevance rating for documents returned
- Comparison of strategy to "expert" strategy
- Overlap of relevance ratings by analysts
- User confidence ratings of findings

Cognitive workload

Cognitive workload ratings (NASA TLX)

Prior and Tacit Knowledge

Efficiency

- Time spent solving problem

Confidence

- User confidence in solution's accuracy

Answer/Report Quality

relationships between problem's entities returned compared with subject's perception of set size

Hypothesis Generation and Tracking

Efficiency

- Time spent solving problem

Accuracy

- Success/failure of problem solution
- Mean score for each system-generated hypothesis (from analysts' ratings)
- Total value for multiple binary scales assessing goodness of system hypothesis
- Analyst's ranking of his hypotheses by importance
- Ranking of system hypotheses
- Number of additional variables considered by system
- Number of variables missed by system
- Percent agreement between system & analyst

Confidence

- User confidence ratings of findings

Cognitive workload

- Cognitive workload ratings (NASA TLX)

Massive Data

Efficiency

- Time to produce findings

Effort

- Number of steps needed to perform a function

Accuracy

- Accuracy of findings in each problem

Confidence

- User confidence ratings of the correctness of his answers

Cognitive workload

- Estimate of task complexity

Human Information Interaction

Efficiency

- time/search
- time/document read

Effort

- # documents accessed
- # documents read
- document growth rate
- document growth type (cut/paste vs. typing)

Accuracy

- Evidence used in analysis
- Number of hypotheses considered
- Average system rank of documents viewed

Confidence

- User confidence ratings of findings

Answer/Report Quality

- Quality of report
- Ranking of report

Cognitive workload

- Cognitive workload ratings (NASA TLX)

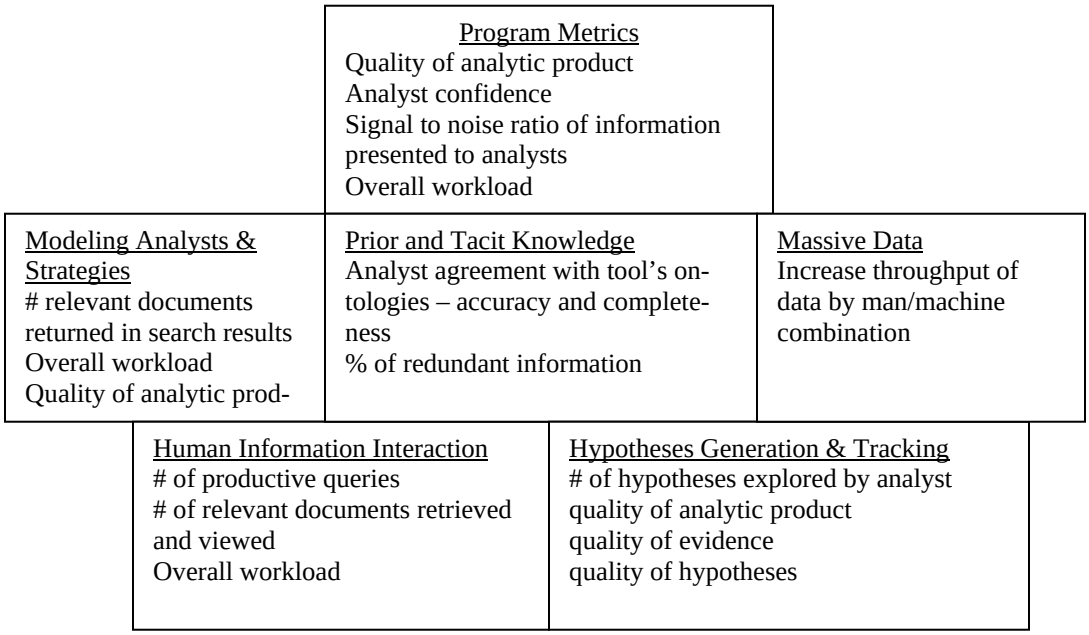

Figure 2: Hierarchy of Metrics

5. Baseline Comparisons

It is important to have a baseline to use in measuring impact. During this project, we have also had between 2 and 4 open source analyst who have been working in the Glass Box environment. In addition to collecting online activities using the Glass Box, we have also been able to conduct observations of these analysts. This has allowed us to verify what, if any, offline activities we have missed [Scholtz, Morse, Hewett 2004]. Additionally, the analysts are able to use an annotation feature of the Glass Box to record offline events. They also often record strategies they are using and give us insights into workshops or other interactions they have had.

The first several sets of this data have been extremely useful to researchers who want to understand the process that the analysts go through.

We are now in the process of an actual baseline study. During this study, two analysts will be working in two separate domains. They will do 6 tasks; each one lasting one week and they will produce a product for each task. These tasks have been carefully designed in conjunction with the tasks we will use in a forthcoming evaluation. That is, we have taken care to design tasks for the baseline study that will be comparable to the tasks we will give to the analysts during the actual software evaluations. It should be noted, however, that task difficulty is a research area in itself. While we do not believe that we have identified all the aspects that contribute to task difficulty, we do think we have at least identified and controlled for some significant aspects.

During the baseline period the analysts have been asked to use the capabilities of the Glass Box to log the relevance of all documents they read. They have also been asked to account for all offline periods of time using the annotation feature. We have asked them to use a format for their reports that lists their assumptions (if any), any hypotheses they investigated, the evidence supporting each of these, and the confidence they have in their recommendations. In addition, each day they will answer a brief debriefing questionnaire and use the NASA TLX to give us an indication of their workload for the day.

Based on this information and additional information from the Glass Box, we are able to calculate:

- Number of searches
- Number of relevant documents
- Number of total documents read
- Growth rate of report
- Number of hypotheses investigated
- Evidence found for each hypotheses
- Percentage of time online/offline

We will conduct several observations during this period to validate the Glass Box data collection.

6. Software Evaluations

We will then start inserting the research software, one tool each month. The analysts will be given training, both on using the features of the software and the optimal use of the software in the analytic process. They will be given one week for training and for experimentation. The second week they will be given a task comparable to one in the baseline period and will be asked to generate a report at the end of the week. The task selected will depend on the capabilities of the particular software. Our initial set of tasks has been designed with knowledge of the research tools that will be used in the evaluations and has been created to take advantage of their capabilities.

We will calculate metrics both from the Glass Box data and from the data that the research applications write to the Glass Box logs. This includes many of the measures in Table 2.

7. Conclusions

We have developed a hierarchy of metrics consisting of those for the overall program and those for the individual research focus areas. A number of the metrics are duplicated in the various research areas. The point is that the impact at the top or program level goals is the sum of the contributions made from the various projects in the focus areas.

While we have not explicitly mapped the lower level metrics into the program metrics in this paper, it should be easy to see that measures, such as number of relevant documents retrieved and number of productive queries, feed into the program level goal of reducing the signal to noise ratio. Measures of the number of hypotheses explored, the amount and quality of the evidence will contribute to a better quality analytic product.

In this paper we have outlined the approach we have taken to developing metrics for intelligence analysts interacting with intelligent software. We presented the metrics and discussed the baseline data we are collecting. We also noted that a number of the research projects will be inserting software into the Glass Box environment for our analysts to work with. Metrics from these trials will be collected in the fall of 2005. By the time of the symposium, we should be able to report on the data from 5 different trials.

Acknowledgments

This work was supported by ARDA Agreements MOD #7157.04 and MOD #7515.04.

References

ARDA NIMD home page, http://www.ic-arda.org/Novel_Intelligence/ Accessed May 23, 2005.

Cowley, P., Nowell, L., and Scholtz, J. 2005. Glass Box: An Instrumented Infrastructure for Supporting Human Interaction with Information. HICSS 38. Jan 3-6. Hawaii

Hart, S.G. and Staveland, L. E. 1988. Development of a NASA-TLX (Task load index): Results of empirical and theoretical research. Hancock, P. and Meshkati, N. (eds.), *Human Mental Workload*, Amsterdam: North-Holland. pp. 139-183.

Morse, E., Steves, M, and Scholtz, J. 2004. Metrics and methods for evaluating technologies for intelligence analysts. IA 2005 Conference. May 2-6, McLean, VA.

Scholtz, J., Morse, E., and Hewett, T. 2004. In-Depth Observational Studies of Professional Intelligence Analysts. Human Performance, Situation Awareness, and Automation Conference, 22-25 March. 2004, Daytona Beach, FL

Scholtz, J. and Steves, M. 2004. A Framework for Real-World Software System Evaluations. Computer-supported Cooperative Work, Chicago, IL, 6-10 November 2004, 600-603.

Steves, M. and Scholtz, J. 2005. A Framework for Evaluating Collaborative Systems in the Real World. HICSS 38. Jan 3-6. Hawaii